Proyecto2-Pokemon

Doggos

09 de diciembre de 2019

Contents:

Nota: Bienvenido a la documentación del Proyecto 2 de la materia de Redes de Computadoras

CAPÍTULO 1

¿Cómo usar?

Nota: Primero inicializamos el servidor, y después los clientes pueden iniciar una conexión

- Para el servidor
  - Si es la primera vez usando el servidor del juego, por favor dirigirse a la carpeta *./Instalaciones* y ejecutar *./make_servidor*
  - En caso contrario: en una terminal, nos situamos en la ubicación del archivo *pokemonServer.py*
  - No necesitamos parámetros extra para ejecutar el servidor. *./pokemonServer.py*
- Para el cliente
  - Si es la primera vez usando un cliente del juego, por favor dirigirse a la carpeta *./Instalaciones* y ejecutar *./make_cliente*
  - En una terminal, nos situamos en la ubicación del archivo *pokemonClient.py*
  - Este programa recibe como parámetros iniciales la dirección IP a través de la cual se quiere conectar, y el puerto. Por lo tanto, ejecutamos de la siguiente manera: *./pokemonClient.py <IP> <port>*

Nota: El puerto ingresado al ejecutar el cliente debe ser el mismo que usa el servidor

CAPÍTULO 2

Programas involucrados

2.1 Cliente Pokemon Go!

Implementación de un cliente para el juego Pokemon Go! e interactúa directamente con el usuario

pokemonClient.**cerrarSesion**(*soc*)
Cierre normal de sesión del usuario.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**displayCatalogo**(*catalogo*)

Imprime en pantalla el catálogo de Pokemones disponibles de manera «amigable».

Parámetros **catalogo** (*List of String*) – Catalogo de Pokemones

Devuelve Nada

pokemonClient.**displayPokedex**(*pokedex*)

Imprime en pantalla el Pokedex de manera «amigable».

Parámetros **pokedex** (*List of String*) – Pokedex de Pokemones

Devuelve Nada

pokemonClient.**login**(*soc*)
Transfiere los datos al servidor para validar el acceso, y cierra el programa si los datos no son válidos.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**main**()
Función principal

pokemonClient.**muestraCatalogo**(*soc*)

Le muestra el catálogo disponible de Pokemones al usuario.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**muestraPokedex**(*soc*)

Muestra el Pokedex del usuario que solicita esta acción al usuario.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**muestraPokemon**(*bytes*)
Despliega el pokemon asignado.

Parámetros **bytes** (*bytearray*) – bytes de la imagen del pokemon a desplegar

Devuelve Nada

pokemonClient.**playPokemon**(*soc*)
Permite que el usuario juegue Pokemon Go.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**printPokemon**()
Imprime en pantalla el logo Pokemon :returns: Nada

pokemonClient.**terminarConTimeout**(*soc*)

Termina la conexión pues el tiempo de espera de la respuesta del Servidor ha excedido.

Parámetros **soc** (*Socket*) – Socket de la conexión

Devuelve Nada

pokemonClient.**terminarConexion**()

Termina la conexion pues el Servidor notifica que el tiempo de espera ha excedido.

Param Nada

Devuelve Nada

2.2 Servidor Pokemon Go!

Implementación de un servidor para el juego Pokemon Go!

pokemonServer.**avisoTimeout**(*connection*)
Manda el mensaje de cierre de sesión al cliente por tiempo de espera excedido.

Parámetros **connection** (*Conexión*) – Conexión entre el cliente y el servidor

Devuelve Nada

pokemonServer.**cerrarSesion**(*connection*)
Cierre de sesión entre el servidor y el cliente al cual le pertenece la conexión.

Parámetros **connection** (*Conexión*) – Conexión entre el cliente y el servidor

Devuelve Nada

pokemonServer.**clientThread**(*connection*, *ip*, *port*, *max_buffer_size=5120*)
Manejador del hilo que sostiene la conexión entre el servidor y un cliente

Parámetros

- **connection** (*Conexión*) – Conexión entre el servidor y el cliente que abrió el hilo
- **ip** (*String*) – Dirección IP de la conexión
- **port** (*Integer*) – Puerto a través del cual el servidor mantiene la conexión con el cliente
- **max_buffer_size** (*Integer*) – Número máximo de bytes que puede recibir en un paquete del cliente

Devuelve Nada

pokemonServer.**getNombrePokemon**(*idPokemon*)
Regresa el nombre del pokemon dado su id.

Parámetros **idPokemon** (*Integer*) – Id del Pokemon a capturado

Devuelve String

pokemonServer.**giveAccess**(*connection*, *max_buffer_size=5120*)
Autentifica a usuarios registrados y proporciona acceso a la ejecución de la aplicación

Parámetros

- **connection** (*Conexión*) – Conexión entre el servidor y el cliente que abrió el hilo
- **max_buffer_size** (*Integer*) – Número máximo de bytes que puede recibir en un paquete del cliente

Devuelve Integer, String -> Valor que representa la correctud del accesso; Cadena que reprensenta al usuario activo.

pokemonServer.**guardaEnPokedex**(*idPokemon*, *user*)

Guarda el pokemon capturado en el pokedex del usuario.

Parámetros

- **idPokemon** (*Integer*) – Id del Pokemon a capturado
- **user** (*String*) – Usuario que capturo

Devuelve Nada

pokemonServer.**main**()
Función principal.

pokemonServer.**muestraCatalogo**(*connection*)
Muestra el catalogo.

Parámetros **connection** (*Conexión*) – Conexión entre el servidor y el cliente

Devuelve Nada

pokemonServer.**muestraPokedex**(*connection*, *user*)
Muestra el Pokedex del usuario.

Parámetros

- **connection** (*Conexión*) – Conexión entre el servidor y el cliente
- **user** (*String*) – Usuario que capturo

Devuelve Nada

pokemonServer.**playPokemonGo**(*connection*, *user*)
Método que simula el comportamiento del juego Pokemon Go.

Parámetros **connection** (*Conexión*) – Conexión entre el servidor y el cliente

Devuelve Nada

pokemonServer.**start_server**()
Inicialización del servidor

Parámetros **ip_dir** (*String*) – Dirección IP del socket al cual se va conectar el servidor

Devuelve Nada

pokemonServer.**terminarConexion**()

Termina la conexion pues el Cliente notifica que el tiempo de espera ha excedido.

Param Nada

Devuelve Nada

Índice de Módulos Python

p

Índice